Fitting log-multiplicative association models

Dessens, J.; Jansen, W.; Luijkx, R.

Published in:
Glim newsletter

Publication date:
1985

GLIM NEWSLETTER
Issue No. 11

ISSN 0269-0772

NP1078

December 1985

Contents

Published Twice Yearly by
the Royal Statistical Society Glim Working Party
and the Numerical Algorithms Group Ltd

Fitting Log-Multiplicative Association Models

J Dessens, W Jansen & R Luijkx
Department of Sociology
University of Utrecht
Heidelberglaan 2
P O Box 80.108
3508 TC Utrecht
The Netherlands

Abstract
This paper presents a series of GLIM macros for the fitting of log-multiplicative models for contingency tables. It is both a comment on and an alternative to the routine published by Breen, the latter being too limited.

Introduction

In a previous GLIM Newsletter, [1] presented a GLIM routine for fitting Goodman's log-multiplicative models.

At about the same time we presented a comparable series of GLIM macros elsewhere [2].

Comparing the two procedures reveals that Breen's routine is more advanced than ours in one aspect, but covers a limited group of models. We developed macros to fit not only the row and column effects model II, as Breen did, but also macros to fit the **equal** row and column effects model II:

$$\log F_{ij} = \lambda + \lambda_i + \lambda_j + \beta u_i v_j \text{ with } u_i = v_i \quad (1)$$

We also wrote macros to standardise the (row and column) scores (zero mean and standard deviation of one).

We have tried to rewrite portions of our GLIM routine, so that they could be simply added to Breen's. We have to conclude that a straightforward extension of his routine to the **equal** row and column effects model II is not possible. There is a simple reason: in order to fit the **equal** row and column effects model II it is necessary to define columns of the design matrix, that are the sums of `F*COLS` and `S*ROWS` (Breen's notation) [1]. A statement such as `$CALC HOMS = F*COLS + S*ROWS` gives `HOMS` as a variate, not as a factor, and we do not see how an **adequate** factorisation can be afforded. This can be illustrated by the following pieces of the design matrix for a 3 $\times$ 3 table:

Row and Column Effects
Model II:

$$
\begin{array}{cccccc}
U_1 & 0 & 0 & V_1 & 0 & 0 \\
U_2 & 0 & 0 & 0 & V_1 & 0 \\
U_3 & 0 & 0 & 0 & 0 & V_1 \\
0 & U_1 & 0 & V_2 & 0 & 0 \\
0 & U_2 & 0 & 0 & V_2 & 0 \\
0 & U_3 & 0 & 0 & 0 & V_2 \\
0 & 0 & U_1 & V_3 & 0 & 0 \\
0 & 0 & U_2 & 0 & V_3 & 0 \\
0 & 0 & U_3 & 0 & 0 & V_3
\end{array}
$$

`= F*COLS = S*ROWS`

Equal Row and Column Effects
Model II ($U_I = V_I$):

$$
\begin{array}{ccc}
2U_1 & 0 & 0 \\
U_2 & U_1 & 0 \\
U_3 & 0 & U_1 \\
U_2 & U_1 & 0 \\
0 & 2U_2 & 0 \\
0 & U_3 & U_2 \\
U_3 & 0 & U_1 \\
0 & U_3 & U_2 \\
0 & 0 & 2U_3
\end{array}
$$

$\neq$ `F*COLS + S*ROWS`

We will present our original macros to fit log-multiplicative models in GLIM. In our macros the columns of the design matrix are calculated one by one. So we can avoid the above mentioned problem. Besides, our routine yields standardised category scores and the correct number of degrees of freedom.

For the general outline of the models we refer to [1,3]. Here we merely point to some fields of application, before the macros.

GLIM Estimation for K I*J Tables

Clogg [4] extends Goodman's models [5,6,7,8] to K-group analysis of I*J tables. Clogg's program ANOASC [9] is designed for this purpose. We will shortly point out how to proceed using our GLIM macros. We show for the sake of exposition how to fit the 'homogeneous' row and column effects. Homogeneous means here: homogeneous (i.e. equal) over k groups. This model is called group analysis because it assumes the same row and column category scores for each group:

$$u_{ik} = u_{i.} \qquad i = 1,2,\dots,\mathrm{I}; \qquad k = 1,2,\dots,\mathrm{K} \tag{2a}$$

$$v_{jk} = v_{j.} \qquad j = 1,2,\dots,\mathrm{J}; \qquad k = 1,2,\dots,\mathrm{K} \tag{2b}$$

and the 'homogeneous' equal row and column effects with the additional restriction:

$$u_{i.} = v_{i.} \qquad i = 1,2,\dots,I; \qquad k = 1,2,\dots,\mathrm{K} \tag{2c}$$

To fit these models the only change is in the baseline model which must include conditional independence of the variables `X` and `Y` given a group variable, say `C`. This can be obtained by replacing the textstring `X + Y` in macro `M` by: `X*C + Y*C`, after having read in the data for `C`.

Clogg [4] also presents the heterogeneous versions of the two models mentioned above. These can be fitted by estimating the model for each group apart and summing the test statistics for a simultaneous test.

Models with Diagonal Effects

In the macros we show how to use GLIM in estimating the 'pure' Model II. Now we turn to the case, where we want to fit additional parameters, for example diagonal parameters. This feature is very desirable in the analysis of mobility tables, for example with the modelling of occupational inheritance. For this purpose the baseline model in macro `M` has to be changed again.

Let us look at an example of a 5*5 table where we want to include diagonal parameters. First we define the effects for the diagonal cells in the following way:

`$CAL DIA = %IF(%EQ(X,Y),X,0) : DIA = DIA + 1 $FACTOR DIA 6` (3)

and then include `DIA` in the macro `M`: `X + Y + DIA`.

In this model a parameter is included for each diagonal cell.

If instead only one additional density level for the diagonal is required a parameter `DIAG` in the baseline model (macro `M`) has to be included. `DIAG` is defined by:

`$CAL DIAG = %IF(%EQ(X,Y),1,0)` (4)

which leads to the baseline model in the macro `M`: `X + Y + DIAG`.

Numerous other possibilities are left to the imagination of the reader.

Macros for Fitting Log-multiplicative Models (5*5 Table)†

Here follows a listing of our macros for fitting log-multiplicative models on the well-known British mobility table [10]. For another analysis of the 5*5 version of this table see also [11].

To use the routine, the following input is necessary:

`$MACRO M <MODEL> $ENDMAC$ #IT1 or #IT2 (#ST1 or #ST2)`

In <`MODEL`> you have to define your baseline model; e.g. the independence model `X + Y`; or the inheritance model `X + Y + DIA`. Use `X` for the row variable, `Y` for the column variable. Do not use any other one-character variable names. Use `#IT1` and `#ST1` for the RC II model resp. `#IT2` and `#ST2` for the **equal** RC II model.

```
$UNITS 25!
$CAL %R = 5 : %C = 5!
$CAL X = %GL(%R,%C) : Y = %GL(%C,1)!
$FACTOR X %R Y %C!
$VARIATE %R R : %C C!

$MACRO INIT!
$CAL R(X) = X : C(Y) = Y!
$OUTPUT!
$FIT #M!
$OUTPUT 5!
$PRINT : 'BASELINE MODEL [' M '] WITH DEVIANCE ' %DV ' AND DF ' %DF :!
$DEL PER PEC!
$CAL %A = %PL + %R $VAR %A PER!
```

† Actually we wrote a Fortran(77) program that generates the series of GLIM – macros for interactive use. This program **CRM** (**CR**eate **M**acros) is available from the authors (in print or on IBM cards). In **CRM** the only question to be answered is that of the number of rows and columns of the table; **CRM** gives as output the macros `INIT`, `COL`, `ROW`, `IT1`, `ST1` and `H1` for non-square tables and the same set plus `HOM`, `IT2`, `ST2` and `H2` for square tables.

```
$CAL %A = %PL + %C $VAR %A PEC!
$CAL PER = 0 : PEC = 0 : %P = %PL!
$CAL %A = 1 : %I = 0 : %K = .01!
$ENDMAC$!

$YVAR F$ERR P$DATA F$READ!
 50   45    8   18    8
 28  174   84  154   55
 11   78  110  223   96
 14  150  185  714  447
  0   42   72  320  411

$MACRO COL!
$CAL C1 = %IF(%EQ(Y,1),R(X),0)!
$CAL C2 = %IF(%EQ(Y,2),R(X),0)!
$CAL C3 = %IF(%EQ(Y,3),R(X),0)!
$CAL C4 = %IF(%EQ(Y,4),R(X),0)!
$CAL C5 = %IF(%EQ(Y,5),R(X),0)!
$FIT #M + C1 + C2 + C3 + C4 + C5!
$EXTRACCT %PE!
$CAL C(Y) = %PE(%P+Y)!
$ENDMAC$!

$MACRO ROW!
$CAL R1 = %IF(%EQ(X,1),C(Y),0)!
$CAL R2 = %IF(%EQ(X,2),C(Y),0)!
$CAL R3 = %IF(%EQ(X,3),C(Y),0)!
$CAL R4 = %IF(%EQ(X,4),C(Y),0)!
$CAL R5 = %IF(%EQ(X,5),C(Y),0)!
$FIT #M + R1 + R2 + R3 + R4 + R5!
$EXTRACT %PE!
$CAL R(X) = %PE(%P+X)!
$ENDMAC$!

$MACRO HOM!
$CAL C1 = R(Y)*(%EQ(X,1)) + R(X)*(%EQ(Y,1))!
$CAL C2 = R(Y)*(%EQ(X,2)) + R(X)*(%EQ(Y,2))!
$CAL C3 = R(Y)*(%EQ(X,3)) + R(X)*(%EQ(Y,3))!
$CAL C4 = R(Y)*(%EQ(X,4)) + R(X)*(%EQ(Y,4))!
$CAL C5 = R(Y)*(%EQ(X,5)) + R(X)*(%EQ(Y,5))!
$FIT #M + C1 + C2 + C3 + C4 + C5!
$EXTRACT %PE!
$CAL C(Y) = %PE(%P+Y)!
$CAL R = (R+C)/2 : C = R!
$ENDMAC$!

$MACRO IT1!
$USE INIT!
$PRINT : 'CONVERGENCE HISTORY:' : 'ITERATION DEVIANCE PEARSON''S'!
```

```
$WHILE %A H1!
$CAL R(X) = PER(X+%P) : C(Y) = PEC(Y+%P)!
$FIT #M + R1 + R2 + R3 + R4 + R5 + C1 + C2 + C3 + C4 + C5!
$PRINT ' READY' :$!
$ENDMAC$!

$MACRO ST1!
$PRINT : ' NORMALISED SCALE VALUES:' :!
$CAL %A = %CU(R)/%R : R = R - %A!
$CAL %A = %CU(C)/%C : C = C - %A!
$CAL %K = %SQRT(%CU(R**2)) : R = R/%K!
$PRINT ' ROW VARIABLE' : $LOOK R!
$CAL %K = %SQRT(%CU(C**2)) : C = C/%K!
$PRINT ' COLUMN VARIABLE' : $LOOK C!
$CALC RC = R(X)*C(Y)!
$OUTPUT $FIT #M + RC $EXTRACT %PE $OUTPUT 5!
$PRINT : ' U(STAR):' $CALC %PE(%PL)$!
$PRINT ' STANDARDISED SCALE VALUES:'!
$PRINT ' ROW VARIABLE' : $CAL R = R*(%SQRT(%R)) $LOOK R!
$PRINT ' COLUMN VARIABLE' : $CAL C = C*(%SQRT(%C)) $LOOK C!
$CALC RC = R(X)*C(Y)!
$OUTPUT $FIT #M + RC $EXTRACT %PE $OUTPUT 5!
$PRINT : ' U(STAR):' $CALC %PE(%PL)$!
$PRINT ' READY' :$!
$ENDMAC$!

$MACRO H1!
$CAL %I = %I + 1!
$OUTPUT!
$USE ROW!
$CAL PER = (PER-%PE)**2 : %B = %CU(PER) : PER = %PE!
$USE COL!
$CAL PEC = (PEC-%PE)**2 : %A = %CU(PEC) : PEC = %PE!
$CAL %A = %GE((%B+%A)/%P,%K)!
$OUTPUT 5!
$PRINT %I %DV %X2!
$ENDMAC$!

$MACRO IT2!
$USE INIT!
$PRINT : 'CONVERGENCE HISTORY' : 'ITERATION DEVIANCE PEARSON''S'!
$WHILE %A H2!
$CAL R(X) = PER(X+%P)!
$PRINT : 'DEGREES OF FREEDOM' %DF :!
$PRINT ' READY' :$!
$ENDMAC$!

$MACRO ST2!
$CAL %A = %CU(R)/%R : %K = %SQRT(%CU((R-%A)**2))!
```

```
$CAL R = R - %A : R = R/%K!
$PRINT ' NORMALISED SCALE VALUES:' $LOOK R!
$CALC RC = R(X)*R(Y)!
$OUTPUT $FIT #M + RC $EXTRACT %PE $OUTPUT 5!
$PRINT : ' U(STAR):' $CALC %PE(%PL)$!
$PRINT :' STANDARDISED SCALE VALUES:'!
$CAL R = R*(%SQRT(%R)) $LOOK R!
$CALC RC = R(X)*R(Y)!
$OUTPUT $FIT #M + RC $EXTRACT %PE $OUTPUT 5!
$PRINT :  ' U(STAR):' $CALC %PE(%PL)$!
$PRINT ' READY' :$!
$ENDMAC$!

$MACRO H2!
$CAL %I = %I + 1!
$OUTPUT!
$USE HOM!
$OUTPUT 5!
$PRINT %I %DV %X2!
$CAL PER = (PER-%PE)**2 : %A = %CU(PER) : PER = %PE!
$CAL %A = %GE(%A/%P,%K)$!
$ENDMAC$!
$PRINT ' READY' :$!
$RETURN
```

References

[1] Breen, R
Fitting nonhierarchical and association log-linear models using GLIM.
Sociological Methods & Research **13**, pp. 77-107, 1984.

[2] UMS (Utrecht Mobility Seminar)
Stratification and mobility in Sweden, France, Great Britain and the Netherlands in the 1970's.
Paper, presented at the annual meeting of the ISA Research Committee on Social Stratification and Mobility, 10-12 September, 1984, Budapest, 1984.

[3] Breen, R
Log-multiplicative models for contingency tables using GLIM.
GLIM Newsletter **10**, pp. 14-19, 1985.

[4] Clogg, C C
Some models for the analysis of association in multiway cross-classifications having ordered categories.
Journal of the American Statistical Assocation **77**, pp. 803-815, 1982.

[5] Goodman, L A
Multiplicative models for the analysis of occupational mobility tables and other kinds of cross-classification tables.
American Journal of Sociology, **84**, pp. 804-819, 1979a.

[6] Goodman, L A
Simple models for the analysis of association in cross-classifications having ordered categories.

Journal of the American Statistical Association **74**, pp. 537-552, 1979b.

[7] Goodman, L A
Association models and canonical correlation in the analysis of cross-classification having ordered categories.
Journal of the American Statistical Association **76**, pp. 320-334, 1981a.

[8] Goodman, L A
Association models and the bivariate normal for contingency tables with ordered categories.
Biometrika **68**, pp. 347-355, 1981b.

[9] Shockey, J W and Clogg, C C
ANOASC: a computer program for the analysis of association in a set of K I-by-J contingency tables.
Population Issues Research Center: Pennsylvania State University. *Journal of the American Statistical Association* **77**, pp. 803-815, 1982.

[10] Glass, D V
Social Mobility in Britain.
London: RKP, 1954.

[11] Goodman, L A
Some multiplicative models for the analysis of cross-classified data.
Proceedings of the Sixth Berkeley Symposium on Mathematical Statistics and Probability, (ed.) L leCam et al. Berkeley: University of Calilfornia Press, pp. 649-696, 1972.